



Adaptive grouping for the higher-order multilevel fast multipole method

Borries, Oscar Peter; Jørgensen, Erik; Meincke, Peter; Hansen, Per Christian

Published in:
Microwave & Optical Technology Letters

Link to article, DOI:
[10.1002/mop.28611](https://doi.org/10.1002/mop.28611)

Publication date:
2014

[Link back to DTU Orbit](#)

Citation (APA):
Borries, O. P., Jørgensen, E., Meincke, P., & Hansen, P. C. (2014). Adaptive grouping for the higher-order multilevel fast multipole method. *Microwave & Optical Technology Letters*, 56(10), 2451-2456.
<https://doi.org/10.1002/mop.28611>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Adaptive Grouping for the Multi-Level Fast Multipole Method

Oscar Borries, *Student Member, IEEE*, Erik Jørgensen, *Member, IEEE*, Peter Meincke, *Member, IEEE*, and Per Christian Hansen

Abstract—An alternative parameter-free adaptive approach for the grouping of the basis function patterns in the Multi-Level Fast Multipole Method is presented, yielding significant memory savings for most discretizations. Results from both a uniformly and non-uniformly meshed scatterer are presented, showing how the technique is worthwhile even for regular meshes, and demonstrating that there is no loss of accuracy in spite of the large reduction in memory and relatively low computational cost.

Index Terms—MLFMM, Higher-Order Discretization, Irregular meshes

I. INTRODUCTION

WHEN solving large-scale electromagnetic scattering problems, where the unknown is the surface current density induced by an incident electromagnetic field on a scatterer, the Multi-Level Fast Multipole Method (MLFMM) [1]–[4] is one of the most powerful methods for speeding up the necessary matrix-vector products involved in an iterative solution.

The MLFMM is a hierarchical algorithm, achieving an asymptotic complexity of $\mathcal{O}(N \log N)$, N being the number of unknowns, by computing interactions between groups of basis functions rather than individual basis functions. The multi-level aspect comes from using a hierarchical grouping to allow interactions over increasing distances to be done by considering increasingly larger groups.

The first step of the MLFMM is the application of a grouping algorithm. The grouping effectively determines the region of validity of the Greens function expansion underlying the MLFMM. Furthermore, the grouping dictates the number of terms needed to represent the functions involved in MLFMM, often called the *bandwidth*, and therefore also the required number of samples of those functions on the unit sphere, called the *sampling rate*. In most implementations [2], [4], this grouping is done by using the Octree algorithm [5]. This is a fast, easily implemented and conceptually simple algorithm, designed in computer graphics to adapt very well to any geometrical shape. In the context of the MLFMM, it also has the extremely important feature of allowing reuse of some of the quantities involved [6].

Oscar Borries is with TICRA, DK-1201 Copenhagen, Denmark and also with the Department of Applied Mathematics and Computer Science, Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark

Erik Jørgensen and Peter Meincke are with TICRA, DK-1201 Copenhagen, Denmark.

Per Christian Hansen is with the Department of Applied Mathematics and Computer Science, Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark

Manuscript received April 19, 2005; revised December 27, 2012.

However, as the discretizations become more irregular, the Octree grouping at the finest level results in excessive sampling rates, since it only considers the center of the geometrical elements, known as patches, in the discretization and the size of the largest patch, rather than taking into account the size and shape of the individual patch. To improve on the performance, particularly for very irregular meshes or for meshes with large patches, we suggest in this paper a method that allows the grouping on the finest level to become completely adaptive to the shape of the patches. While this method is conceptually simple, it has to our knowledge not been published previously.

The time factor $e^{j\omega t}$, where ω is the angular frequency, is assumed and suppressed throughout.

II. MULTI-LEVEL FAST MULTIPOLE METHOD

In the present paper, the MLFMM is used when iteratively solving the Electric Field Integral Equation (EFIE) [7]

$$\hat{\mathbf{n}} \times \mathbf{E}^i = \mathcal{L} \mathbf{J}_S, \quad (1)$$

where $\hat{\mathbf{n}}$ is a unit vector normal to the scatterer \mathcal{S} , \mathbf{E}^i is the incident electric field, and \mathbf{J}_S is the unknown surface current density. Further, \mathcal{L} is the integral operator

$$\mathcal{L} \mathbf{J}_S = \hat{\mathbf{n}} \times j\omega\mu \left[\int_{\mathcal{S}} \mathbf{J}_S(\mathbf{r}') G(\mathbf{r}, \mathbf{r}') d^2 \mathbf{r}' + \frac{1}{k^2} \int_{\mathcal{S}} \nabla'_{\mathcal{S}} \cdot \mathbf{J}_S(\mathbf{r}') \nabla G(\mathbf{r}, \mathbf{r}') d^2 \mathbf{r}' \right], \quad (2)$$

where μ is the free-space permeability, $k = 2\pi/\lambda$, with λ being the free-space wavelength. $G(\mathbf{r}, \mathbf{r}') = \frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r}-\mathbf{r}'|}$ is the free-space Greens function and \mathbf{r} , \mathbf{r}' denote observation and integration points, respectively. For some scenarios, it is more useful to apply the Combined Field Integral Equation (CFIE) [7],

$$\left[\alpha \mathcal{L} + (1 - \alpha) \eta \left(\frac{1}{2} \mathcal{I} + \mathcal{K} \right) \right] \mathbf{J}_S = \alpha \hat{\mathbf{n}} \times \mathbf{E}^i + (1 - \alpha) \eta \hat{\mathbf{n}} \times \mathbf{H}^i. \quad (3)$$

where \mathcal{I} is the identity operator, \mathbf{H}^i is the incident magnetic field, η is the free-space impedance, $\alpha \in [0, 1]$ is a weighting factor, and \mathcal{K} is the operator

$$\mathcal{K} \mathbf{J}_S = \hat{\mathbf{n}} \times \oint_{\mathcal{S}} \mathbf{J}_S(\mathbf{r}') \times \nabla G(\mathbf{r}, \mathbf{r}') d^2 \mathbf{r}', \quad (4)$$

where \oint denotes the Cauchy principal value.

The central part of the MLFMM is Rokhlin's translator [1]

$$T_L(k, \hat{\mathbf{k}}, \mathbf{x}) = \sum_{l=0}^L (-j)^l (2l+1) h_l^{(2)}(k \|\mathbf{x}\|_2) P_l(\hat{\mathbf{k}} \cdot \hat{\mathbf{x}}), \quad (5)$$

where $\hat{\mathbf{k}}$ is the unit wave vector, \mathbf{x} is the vector between two group centers, $\hat{\mathbf{x}} = \mathbf{x}/\|\mathbf{x}\|_2$, $h_l^{(2)}$ is the spherical Hankel function of second kind and order l , and P_l is the Legendre polynomial of degree l . It is important to note that the translator does not depend on the absolute position of the groups, but only on the vector \mathbf{x} between their centers. Thus, the translator can be reused for pairs of groups with the same \mathbf{x} , a key factor in keeping memory consumption low. Typically, the number of terms $L+1$ in the translator is determined from the Excess Bandwidth Formula [8]

$$L = kD + 1.8\beta^{2/3}(kD)^{1/3}, \quad (6)$$

where $D = \sqrt{3}a$ is the diameter of the group, a is the sidelength of the group, and $10^{-\beta}$ is the desired relative error.

To discretize the problem, we begin by representing the surface of the scatterer \mathcal{S} by geometric elements known as patches. Then, (1) is discretized using a Galerkin method to yield a linear system $\bar{\bar{\mathbf{Z}}}\bar{\mathbf{I}} = \bar{\mathbf{B}}$. Here, $\bar{\mathbf{I}}$ is a vector containing coefficients to the basis functions expressing the surface current density, $\bar{\bar{\mathbf{Z}}}$ is a matrix containing as its (i, j) 'th component the mutual impedance between basis functions \mathbf{f}_i and \mathbf{f}_j , and $\bar{\mathbf{B}}$ is a vector, representing the incident field as tested by the basis functions [9]. We can consider MLFMM as a method for splitting the matrix $\bar{\bar{\mathbf{Z}}}$ into two parts

$$\bar{\bar{\mathbf{Z}}} = \bar{\bar{\mathbf{Z}}}_{\text{near}} + \bar{\bar{\mathbf{Z}}}_{\text{far}}, \quad (7)$$

where the *near-matrix* $\bar{\bar{\mathbf{Z}}}_{\text{near}}$ is stored as a sparse matrix, while $\bar{\bar{\mathbf{Z}}}_{\text{far}}$ is not stored directly, but instead the elements required to multiply $\bar{\bar{\mathbf{Z}}}_{\text{far}}$ with an excitation vector $\bar{\mathbf{I}}$ are stored.

Introducing the basis function patterns as

$$\mathbf{V}_{jm}(k, \hat{\mathbf{k}}) = \int_{\mathcal{S}} \mathbf{f}_j(\mathbf{r}) \cdot [\bar{\mathbf{I}} - \hat{\mathbf{k}}\hat{\mathbf{k}}] e^{-jk\hat{\mathbf{k}} \cdot (\mathbf{r}_m - \mathbf{r})} d^2\mathbf{r}, \quad (8)$$

where $\bar{\mathbf{I}}$ is the identity matrix, and utilizing (5), we can express the matrix elements resulting from the EFIE (1) as

$$\bar{\bar{\mathbf{Z}}}_{\text{far}(j,i)} = \oint \mathbf{V}_{jm}^*(k, \hat{\mathbf{k}}_p) \cdot \left(T_L(k, \hat{\mathbf{k}}, \mathbf{r}_{mm'}) \mathbf{V}_{im'}(k, \hat{\mathbf{k}}) \right) d^2\hat{\mathbf{k}}, \quad (9)$$

which is then discretized to

$$\bar{\bar{\mathbf{Z}}}_{\text{far}(j,i)} = \kappa \sum_{p=1}^K w_p \mathbf{V}_{jm}^*(k, \hat{\mathbf{k}}_p) \cdot \left(T_L(k, \hat{\mathbf{k}}_p, \mathbf{r}_{mm'}) \mathbf{V}_{im'}(k, \hat{\mathbf{k}}_p) \right), \quad (10)$$

where κ is a constant depending on the units of the impedance matrix $\bar{\bar{\mathbf{Z}}}$, $K = 2(L+1)^2$ is the number of sample points on the unit sphere [10], and w_p are the integration weights. We assume that \mathbf{f}_i belongs to group m' and \mathbf{f}_j belongs to group m , $\mathbf{r}_{mm'} = \mathbf{r}_m - \mathbf{r}_{m'}$ where \mathbf{r}_m denotes the center of group m , and we further assume that $|\mathbf{r}_{mm'}| > D$. If $|\mathbf{r}_{mm'}| \leq D$, the element $\bar{\bar{\mathbf{Z}}}_{j,i}$ must be computed directly and stored in $\bar{\bar{\mathbf{Z}}}_{\text{near}}$. For the CFIE (3), the expression for the matrix elements corresponding to (10) are slightly more complicated [4].

The key issue from (8) and (10) is that the number of sample points for the basis function patterns \mathbf{V}_{jm} is the same as that required for the translator, even though the bandwidth of \mathbf{V}_{jm} is lower [10]. Note that the bandwidth of \mathbf{V}_{jm} is directly related to the largest value of $|\mathbf{r}_m - \mathbf{r}|$ attained on the domain of \mathbf{f}_j , due to the term $e^{-jk\hat{\mathbf{k}} \cdot (\mathbf{r}_m - \mathbf{r})}$ in (8).

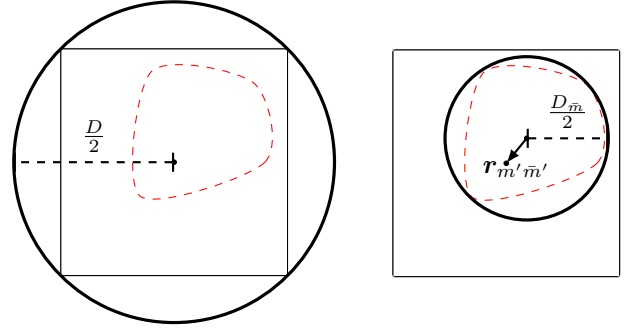


Fig. 1. 2-D illustration of adaptive grouping. The dashed line is the projection of a patch onto a plane, while the grey square is the box at the finest level of the Octree. To the left is shown the scenario obtained with using the Octree grouping at the finest level as a foundation for the basis function patterns. Further subpartitioning would dissect the patch, which is suboptimal. Instead, on the right, we introduce an adaptive grouping layer, which has its center such as to minimize the size of the circle enclosing the patch. We thus see that the region of validity, indicated by the solid black circle, is much smaller than it would be if it had to enclose the entire square. The $\mathbf{r}_{m'm'}$ vector indicates the phase-center shift needed to start the upward pass.

III. GROUPING

An Octree [5] is used here as a hierarchical data structure, allowing a geometrical object to be spatially partitioned in a fast and simple manner, thereby providing a grouping of basis functions that are spatially near each other. The Octree grouping is done by creating an original bounding box for \mathcal{S} , hereafter termed level 1 or the coarsest level. Finer levels are then created by partitioning boxes such that the diameter of the boxes at level q is $D_q = D_{q-1}/2$. This results in eight potential boxes per partitioning, of which only those that contain the center of a patch are kept while the rest are pruned. This results in a very fast partitioning of the patches into clusters. The partitioning stops when

$$D_q/2 < l, \quad (11)$$

where l is the largest sidelength of any patch in the mesh surface, yielding q levels in the Octree.

However, as Figure 1 illustrates, this will occasionally result in unnecessarily large groups at the finest level, simply because the Octree scheme is not able to adapt to the patches. Effectively, the center point for \mathbf{V}_{jm} , \mathbf{r}_m , is positioned such that the sample rate will be far too large. The memory cost for this can be very significant. Further, since the sampling rate for all boxes at each level is the same, and since the finest level box size is determined by the largest patch length in the mesh, scenarios with non-uniform patch sizes will result in far too large sampling rates for the groups with smaller patches.

Therefore, we propose to tabulate each \mathbf{V}_{jm} based on a separate *adaptive grouping*. In this approach, each patch is associated with its own group, with the center point \mathbf{r}_m chosen to minimize the term $|\mathbf{r}_m - \mathbf{r}|$ in (8). D is then found by the maximum attained value of $2|\mathbf{r}_m - \mathbf{r}|$, and the sampling rate is determined from (6). In this way, the sampling rate is optimized for each patch, and the basis function patterns are stored at the coarsest possible sampling density.

With this adaptive grouping, (10) is changed to

$$\bar{\bar{Z}}_{j,i} = \kappa \sum_{p=1}^K w_p \mathbf{V}_{j\bar{m}}^*(k, \hat{\mathbf{k}}_{\bar{p}}) \cdot \left(W^T e^{-jk\hat{\mathbf{k}} \cdot \mathbf{r}_{\bar{m}}} \cdot T_L(k, \hat{\mathbf{k}}_{\bar{p}}, \mathbf{r}_{m'm'}) e^{-jk\hat{\mathbf{k}} \cdot \mathbf{r}_{m'\bar{m}'}} W \mathbf{V}_{i\bar{m}'}(k, \hat{\mathbf{k}}_{\bar{p}}) \right), \quad (12)$$

where the notation \bar{m} refers to a group at the adaptive level, and group m is the group at the finest level of the Octree containing \bar{m} . W is an interpolation matrix, designed to increase the sampling density of the basis function patterns to that of the translator, such that $W \in \mathbb{R}^{K \times \bar{K}}$.

We stress that there are no translations done on the adaptive grouping level, and therefore the near-matrix $\bar{\bar{Z}}_{\text{near}}$ is not based on the adaptive level. Basing $\bar{\bar{Z}}_{\text{near}}$ on the adaptive level would yield a smaller matrix, but this would imply that separate translators would have to be computed for each adaptive group interaction due to the adaptive group center. Without this possibility of reusing translators, which is perhaps the greatest strength of the Octree used with the MLFMM, the memory requirements for the translators, as well as the additional work in translation on the adaptive level, would impair the performance. Another advantage in using the larger Octree groups for the translation is that (5) is more numerically stable for larger groups due to the so-called sub-wavelength breakdown [11].

In a practical implementation, if each adaptive group has a unique diameter, and thus a unique sampling rate, this would require a significant amount of auxiliary data, in particular the interpolation matrix W for each group. Therefore, our implementation uses a number of specifically allowed sizes, and categorizes each adaptive group into those. Note that only the sampling rate is affected by this categorization, not the center of the adaptive group. For fairly uniform scatterers, only 2 or 3 possible sizes are needed, while strongly non-uniform scatterers need a few more. As an estimate for the number of allowed group sizes $N_{\bar{m}}$, we use

$$N_{\bar{m}} = \left\lceil \frac{\max_{\bar{m}} D_{\bar{m}}}{\min_{\bar{m}} D_{\bar{m}}} \right\rceil, \quad (13)$$

where \bar{m} runs through the adaptive groups, and $D_{\bar{m}}$ is the diameter of group \bar{m} .

The adaptive grouping does not affect the number of non-zeros in the near-matrix $\bar{\bar{Z}}_{\text{near}}$. To reduce the memory required for $\bar{\bar{Z}}_{\text{near}}$, a locally extended Octree grouping can be implemented. This means that l in (11) is modified to equal the largest sidelength of any quad in the group under consideration only, rather than considering all quads in the mesh. Thus, if the mesh is locally very fine, the Octree will locally have additional levels compared to regions with a coarser mesh. Figure 2 provides a small 2-D illustration of a locally extended Octree grouping.

A locally extended Octree was discussed in [12], but was used as a way to reduce the size of the basis function patterns only, not to reduce the memory required for $\bar{\bar{Z}}_{\text{near}}$, since translations on the extended levels was not performed. Further, the focus was not on irregular meshes, so neither the approach nor the results can be compared with the present paper.

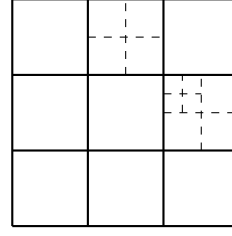


Fig. 2. 2-D illustration of an Octree grouping using locally extended levels. The solid lines indicate the finest level achieved using the standard Octree grouping. The dashed lines indicate regions where the Octree has been locally refined due to small patches being present in the groups.

IV. NUMERICAL RESULTS

The first example involves a perfectly electrically conducting (PEC) sphere, designed to illustrate that even for uniformly meshed scatterers, it is beneficial to apply the adaptive grouping, particularly if the group size on the finest level of the Octree is much larger than the patch size. It further illustrates that there is no loss of accuracy from the adaptive grouping.

The second example concerns an irregularly meshed circular PEC plate with several small holes, designed to represent mounting holes. While we stress that these holes are so small that they should not be considered in the electromagnetic representation of the problem, it is fairly common in structures based on CAD designs to have such features.

The results are based on the implementation detailed in [13], but the implementation does not utilize the storage of basis functions using Spherical Harmonics Expansions (SHE) [14], since we want to isolate the effects of using adaptive grouping compared to standard Octree grouping at the finest level. However, these two techniques (adaptive grouping and SHE) can easily be combined, and their combination allows use of the SHE to reduce the computational cost of adaptive grouping. We use Lagrange interpolators to step between the sampling rates of the levels. When discussing total memory, we include the memory needed to store the entire MLFMM structure, including near-matrix, basis function patterns etc., as well as minor temporary data, including that needed for interpolation matrices in the adaptive grouping. Throughout, the accuracy setting $\beta = 3$ is used in (6). The error is computed as the relative RMS

$$\text{Relative RMS} = \sqrt{\frac{\sum_{i=1}^{N_s} (|\mathbf{E}_{i,\text{ref}}| - |\mathbf{E}_{i,\text{cal}}|)^2}{\sum_{i=1}^{N_s} |\mathbf{E}_{i,\text{ref}}|^2}}, \quad (14)$$

where $\mathbf{E}_{i,\text{ref}}$ and $\mathbf{E}_{i,\text{cal}}$ denote the reference and calculated electric fields at the i^{th} sample point, respectively, and N_s is the number of samples.

A. Sphere

We consider an \hat{x} -polarized plane wave at 10 GHz, propagating in the $+\hat{z}$ direction, incident on a 1 m PEC sphere centered at the origin of the coordinate system. Using 4th-order basis functions, we vary the sidelength of the patches in the fairly uniformly meshed structure, and use the CFIE (3) with

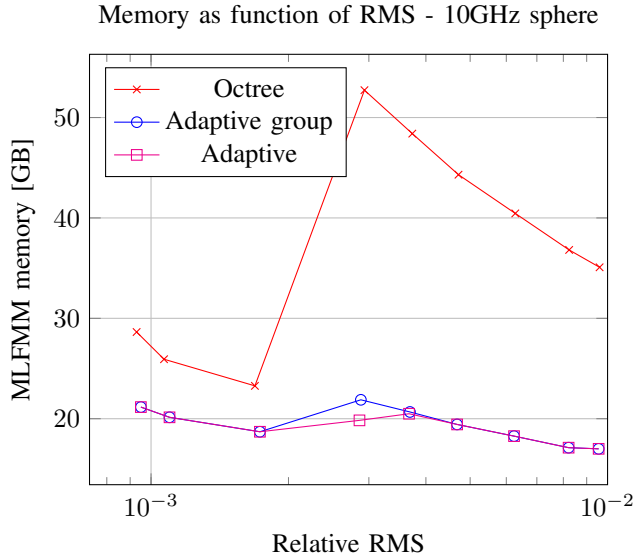


Fig. 3. Memory for the entire MLFMM structure, as a function of the accuracy, comparing the traditional Octree grouping at the finest level, using adaptive grouping only, and using adaptive grouping as well as the locally extended Octree ("Adaptive"). The RMS is increased by increasing the patch length.

$\alpha = 0.5$ to solve the problem, requiring between 322752 and 668352 unknowns. Figure 3 shows the memory consumption as function of the RMS, illustrating the unfortunate property of the Octree grouping to have a very complicated dependence between memory and RMS. In particular, for the scenario $D_q/2 < l \ll D_q$, where l is the largest sidelength in the mesh and q is the finest level, the standard Octree grouping results in extreme memory consumption for the basis function patterns, visible as a peak Figure 3. The adaptive grouping does not have this problem for the basis function patterns, though we can still see the effects through the memory used for the near-matrix. We further see that including the locally extended Octree only has an effect at a single point, further smoothing out the "hump" when the discretization is getting close to allowing an additional level in the Octree.

Figure 4 illustrates the time spent per matrix-vector product. We see a modest increase from using the adaptive grouping due to the extra interpolation step. However, particularly for Higher-Order discretizations, where there are relatively few groups at the finest level, and thus fewer interpolation steps, this will be negligible compared to the significant reduction in memory. Thus, we can conclude that even for uniformly meshed scatterers, there is a very significant potential memory reduction to be achieved by using adaptive grouping at the cost of a modest increase in computational time.

B. Circular plate with holes

We now consider an \hat{x} -polarized plane wave at 300 MHz, propagating along $-\hat{z}$, incident on a 36 m diameter circular plate, centered at the origin and positioned in the xy -plane. The plate has 9 square mounting holes, each with a sidelength of 0.1λ , placed in a cross around the center. The meshing of this surface with quadrilaterals using sidelengths between 0.1λ

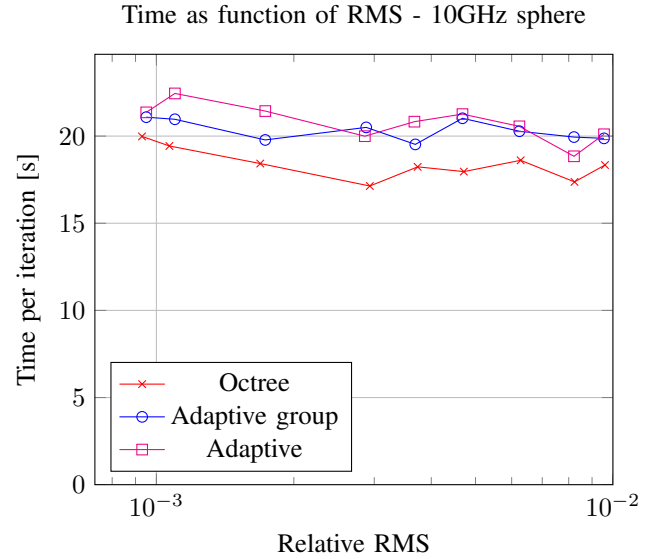


Fig. 4. Time per matrix-vector product, as a function of the accuracy, comparing the traditional Octree grouping at the finest level, using adaptive grouping only, and using adaptive grouping as well as the locally extended Octree ("Adaptive"). We note that the differences between the adaptive grouping only and the "adaptive" approach is predominantly due to inaccuracies in the timings.

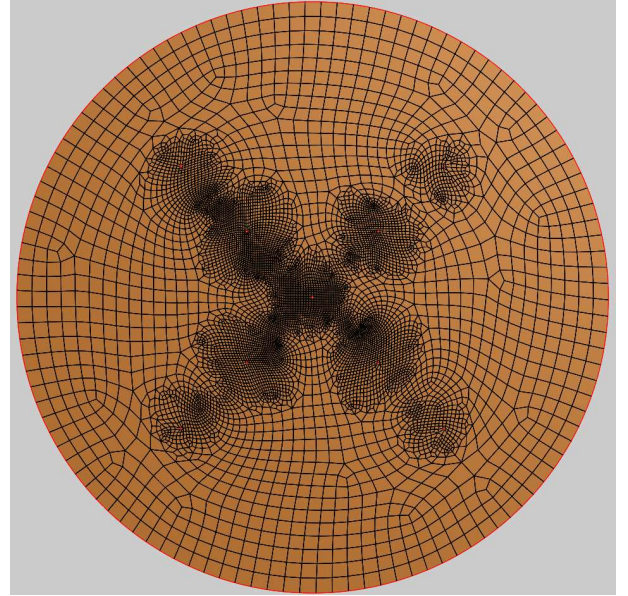


Fig. 5. Circular plate with nine $0.1\lambda \times 0.1\lambda$ mounting holes.

and 1.28λ is shown in Figure 5.

The discretization yields $N = 51491$ unknowns, with polynomials up to 7th order being used on the largest patches. The default Octree grouping yields a fairly poor grouping, with a sidelength at the finest level of 2.25λ . This results in 3.79 GB of total memory. Using the adaptive grouping and locally extended Octree, the memory is reduced to less than a sixth, 623 MB. The memory for the basis function patterns alone is reduced from 3.1 GB to 426 MB, a factor of 7.5. We note that with adaptive grouping, but without the locally extended Octree, the total memory required would have been 1.09 GB,

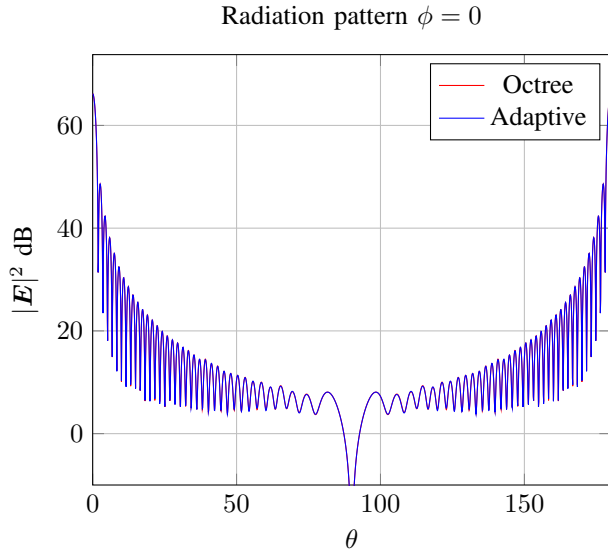


Fig. 6. The scattered far-field for a circular plate with holes, for $\phi = 0$. Both the results from using Octree grouping and the adaptive grouping is shown, demonstrating that there is no loss of accuracy.

so for this strongly non-uniform mesh, the locally extended Octree is effective.

Figure 6 shows the scattered fields from each of the two techniques, further demonstrating that the proposed adaptive approach, with both adaptive grouping and a locally extended Octree, yields the same result as the standard Octree grouping.

V. CONCLUSION

Our results demonstrate that the proposed adaptive grouping approach should be included in modern implementations of MLFMM, particularly when using a Higher-Order discretization with larger patch sidelengths. Further, for strongly non-uniform meshes, a locally extended Octree should also be implemented. For a modest increase in computational time, the reduction in memory obtained with these methods is significant and the implementation is simple.

REFERENCES

- [1] V. Rokhlin, "Diagonal Forms of Translation Operators for Helmholtz Equation in Three Dimensions," Tech. Rep., Mar. 1992.
- [2] C.-C. Lu and W. C. Chew, "A Multilevel Algorithm for Solving a Boundary Integral Equation of Wave Scattering," *Microwave and Optical Technology Letters*, vol. 7, no. 10, Jul. 1994.
- [3] R. Coifman, V. Rokhlin, and S. M. Wandzura, "The Fast Multipole Method for the Wave Equation: A Pedestrian Prescription," *IEEE Antennas and Propagation Magazine*, 1993.
- [4] J. M. Song and W. C. Chew, "Multilevel fast-multipole algorithm for solving combined field integral equations of electromagnetic scattering," *Microwave and Optical Technology Letters*, vol. 10, no. 1, pp. 14–19, Sep. 1995.
- [5] D. Meagher, "Geometric modeling using octree encoding," *Computer Graphics and Image Processing*, vol. 19, no. 2, pp. 129–147, Jun. 1982.
- [6] S. Velampambal, W. C. Chew, and J. Song, "10 Million Unknowns: Is It That Big?" *IEEE Antennas and Propagation Magazine*, vol. 45, no. 2, pp. 43–58, Apr. 2003.
- [7] J. R. Mautz and R. F. Harrington, "H-Field, E-Field, and Combined Field Solutions for Bodies of Revolution," *Syracuse University, Department of Electrical and Computer Engineering*, Mar. 1977.

- [8] J. Song and W. C. Chew, "Error Analysis for the Truncation of Multipole Expansion of Vector Green's Functions," *IEEE Microwave and Wireless Components Letters*, vol. 11, no. 7, pp. 311–313, Jul. 2001.
- [9] R. F. Harrington, "Matrix methods for field problems," *Proceedings of the IEEE*, vol. 55, no. 2, pp. 136–149, 1967.
- [10] S. Koc, J. M. Song, and W. C. Chew, "Error Analysis for the Numerical Evaluation of the Diagonal Forms of the Scalar Spherical Addition Theorem," *SIAM Journal on Numerical Analysis*, vol. 36, no. 3, Mar. 1999.
- [11] M. L. Hastriter, S. Ohnuki, and W. C. Chew, "Error control of the translation operator in 3D MLFMA," *Microwave and Optical Technology Letters*, vol. 37, no. 3, pp. 184–188, May 2003.
- [12] X.-M. Pan, L. Cai, and X.-Q. Sheng, "An Efficient High Order Multilevel Fast Multipole Algorithm for Electromagnetic Scattering Analysis," *Progress In Electromagnetics Research*, vol. 126, pp. 85–100, 2012.
- [13] O. Borries, P. Meincke, E. Jørgensen, and P. C. Hansen, "Multi-level Fast Multipole Method for Higher-Order Discretizations," *IEEE Transactions on Antennas and Propagation*, submitted 2013.
- [14] T. F. Eibert, "A diagonalized multilevel fast multipole method with spherical harmonics expansion of the k-space Integrals," *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 2, pp. 814–817, Feb. 2005.